



Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analogico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

Aulas 01 e 02 - Introdução, Sistemas de Numeração e Codificação de Dados

Hugo Vinícius Leão e Silva

hugovlsilva@gmail.com, hugo.vinicius.16@gmail.com, hugovinicius@ifg.edu.br

Instituto Federal de Educação, Ciência e Tecnologia de Goiás
Campus Anápolis
Curso de Bacharelado em Ciência da Computação

16 de setembro de 2021



Visão geral

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analogico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- 1 Representações de sinais
- 2 Sistemas digitais e analógicos
- 3 Representando bits
- 4 Circuitos lógicos
- 5 Modos de transmissão
- 6 Memória
- 7 Computadores digitais
- 8 Sistemas de numeração
- 9 Codificação/decodificação de dados



Representações de sinais

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- Maioria dos dispositivos eletrônicos atuais é digital e binária;
- Nesse contexto, o outro lado são os dispositivos analógicos;
- Relembrando: sinais são medições de fenômenos que ocorrem na natureza;
- Um sinal analógico o valor de uma medição é proporcional ao que está sendo medido – velocidade do carro, temperatura ambiente etc;
- Então, um sinal analógico varia continuamente dentro de uma faixa de valores;



Representações de sinais

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- Por outro lado, no sinal digital, os valores são dados por símbolos/dígitos – número de passos, quantidade de pessoas etc;
- Assim, um sinal digital varia discretamente dentro de uma faixa de valores;
- Exemplo no MATLAB `ex1_tiposSinais.m`.

- Um sistema digital processa informações lógicas e sinais digitais;
- Normalmente são dispositivos eletrônicos – sistemas telefônicos, de TV e rádio modernos, de transmissão de dados, computador etc;
- Um sistema analógico processa sinais analógicos. Normalmente são dispositivos eletrônicos, mas podem ser mecânicos, magnéticos, pneumáticos, hidráulicos etc – sistemas telefônicos, de TV e rádio legados, leitores/gravadores de áudio em fita magnética e disco de vinil, *dimmer*;



Por que digital?

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- 1 Sistemas digitais são mais fáceis de serem projetados – o valor *exato* da tensão (aka “voltagem”) ou corrente elétrica (“amperagem”) do sinal não é tão importante, apenas o valor aproximado.
- 2 É muito mais fácil armazenar sinais digitais do que suas as contrapartes analógicas;
- 3 Também é mais fácil dimensionar a precisão de um sistema digital – um exemplo: redimensionar uma ULA projetada para float para double. Sistemas analógicos possuem em geral pouca precisão, até três ou quatro casas decimais. Float tem até oito casas decimais e double, dezesseis;



Por que digital?

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- 4 Sistemas digitais podem ser programados;
- 5 Por causa do item 1, o ruído afeta *menos* um sistema digital – exemplo no MATLAB `ex2_ruídoSinais.m`;
- 6 É possível aumentar o nível de integração nos circuitos integrados – mais componentes dentro de um *chip* sem aumentar o tamanho;
- 7 Observa-se uma tendência cada vez maior da *digitalização* de sistemas legados.

- 1 O mundo é analógico** – a maioria das medições de fenômenos na natureza são inerentemente analógicas. Assim, na digitalização desses sinais, sempre haverá perdas. Exemplo trivial: temperatura no IFG agora é $24,4^{\circ}$, mas um sistema digital *poderia* medir apenas 24° ;
- 2** A digitalização (conversão analógico-digital ou conversão A/D) e a conversão D/A levam tempo e adicionam complexidade ao sistema em relação a um sistema puramente digital – exemplo: assistir a um jogo de futebol na TV analógica e na TV digital;
- 3** Há situações em que técnicas analógicas são melhores do que as suas contrapartes digitais – exemplo: amplificação de sinais;

Mesmo assim, as vantagens dos sistemas digitais sobre os analógicos superam largamente as suas deficiências

- Assim, o melhor é utilizar sistemas híbridos que fazem uso das vantagens dos sinais analógicos e digitais;
- Exemplos: CDs, DVDs e Blu-rays têm uma qualidade de áudio e vídeo muito melhor a um custo menor do que os sistemas legados (vinil, fita cassete, VHS);
- Nota-se cada vez mais *digitalização* de sistemas;

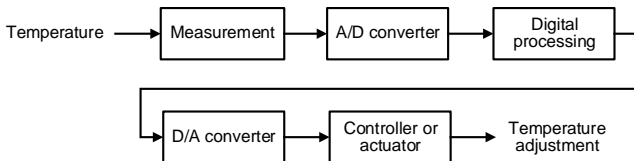
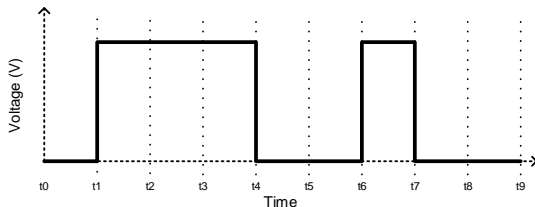


Figura: Diagrama de blocos de um sistema de controle de temperatura

- Normalmente, sistemas digitais utilizam informações binárias;
- *Bits* são representados por qualquer coisa que tenha dois estados – exemplos: aberto (0) ou fechado (1); papel perfurado (1) ou não (0) etc;
- Nos sistemas digitais, os *bits* são representados por tensão (V) ou corrente elétrica (A);
- No entanto, esses valores não precisam ser exatos exemplo: entre 0 V e 0,8 V (0) e entre 2 V e 5 V (1).



Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- A lógica de um circuito descreve a reação de um circuito digital a determinada entrada;
- Ou seja: Circuitos Digitais == Circuitos Lógicos;
- A maioria dos circuitos digitais são implementados em circuitos integrados (aka *chips*);
- Embora não seja nossa preocupação, as tecnologias mais utilizadas são: TTL (*Transistor-Transistor Logic*) e CMOS (*Complementary Metal Oxide Semiconductor*).

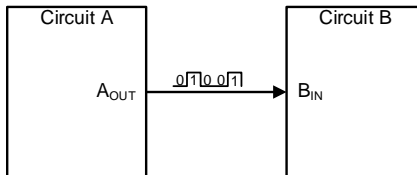


Figura: Transmissão serial

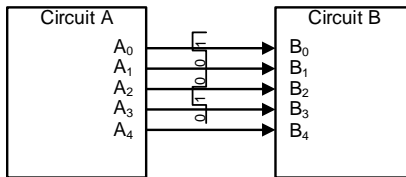


Figura: Transmissão paralela

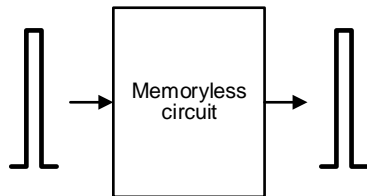


Figura: Dispositivo sem memória

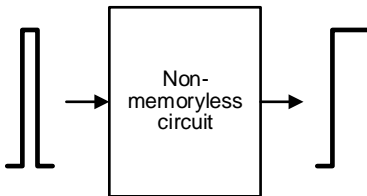


Figura: Dispositivo com memória

- Os computadores digitais são capazes de realizar operações aritméticas, manipular dados geralmente binários e tomar decisões;
- $\text{Computador} == \text{Un. Entrada} + \text{Un. Memória} + (\text{Un. Controle} + \text{Un. Lógico-Aritmética}) + \text{Un. Saída}$;
- $(\text{Un. Controle} + \text{Un. Lógico-Aritmética}) == \text{CPU (ou microprocessador)}$
- Nos computadores de propósito-geral atuais, a CPU está em um *chip*;
- Computadores mais especializados possuem microcontroladores $== \text{CPU} + \text{Entrada} + \text{Saída} + \text{Memória}$;
- São usados em máquinas, videocassetes e *appliances* em geral, ATMs, fotocopiadoras, injeção eletrônica etc.

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

Decimal	Binary	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

- Como converter de base-10 para base-2 ou base-8 ou base-16?
- Como converter de base-2 ou base-8 ou base-16 para base-10?
- Como converter de base-8 ou base-16 para base-2?
- Como converter de base-2 para base-8 ou base-16?
- *The Least Significant Bit (LSB)?*
- *The Most Significant Bit (MSB)?*
- *Bit, byte, nibble?*
- Qual a faixa de valores representados por N bits?
- Soma, subtração e multiplicação binária?

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

- No semestre passado vimos as codificações básicas para:
 - Texto – ASCII (*American Standard Code for Information Interchange*) e Unicode;
 - Imagens e vídeo – RGB (*Red, Green, Blue*);
 - Áudio – PCM (*Pulse Code Modulation*);
 - Números – sinal-magnitude, complemento-de-um, complemento-de-dois → Aprendemos que nem sempre é utilizada a representação binária pura para números.

- Além do SM, CD1 e CD2, também existe a BCD – *Binary-Coded Decimal*;
- BCD é um misto entre binário e decimal puros;
- Cada dígito decimal é codificado com os quatro bits correspondentes da conversão dec \rightarrow bin (de 0, 0000, a 9, 1001);
- As sequências 1110 a 1111 não são utilizadas;
- O BCD não é um sistema de numeração como os vistos no semestre passado, exemplo:
 - $(137)_{10} == (1000\ 1001)_2$ (binário);
 - $(137)_{10} == 0001\ 0011\ 0111$ (BCD).
- Logo se percebe que o BCD é menos eficiente.

- Durante o processo de transferência, leitura ou gravação de dados *podem* ocorrer erros;
- Esses erros são, em geral, causados por ruído elétrico – flutuações espúrias de tensão ou corrente – embora também ser causados por interferência;
- Exemplo no MATLAB `ex3_ruídoSinalDigital.m`
- Muitos equipamentos são projetados para serem razoavelmente livres de erros;
- Isso é possível com o uso de códigos detectores de erros (EDCs, do inglês *Error-Detecting Codes*);
- Atualmente, equipamentos transmitem milhões (Mbps), bilhões (Gbps) e até trilhões (Tbps) de bits por segundo. Mesmo uma baixa BER (*Bit Error Rate*, Taxa de Erro de Bit) pode gerar apenas um incômodo ou um desastre.

Aulas 01 e 02

Hugo Silva

Sinais

Digital e
Analógico

Representando
bits

Circuitos
lógicos

Modos de
transmissão

Memória

Computadores
digitais

Sis.
numeração

Codec de
dados

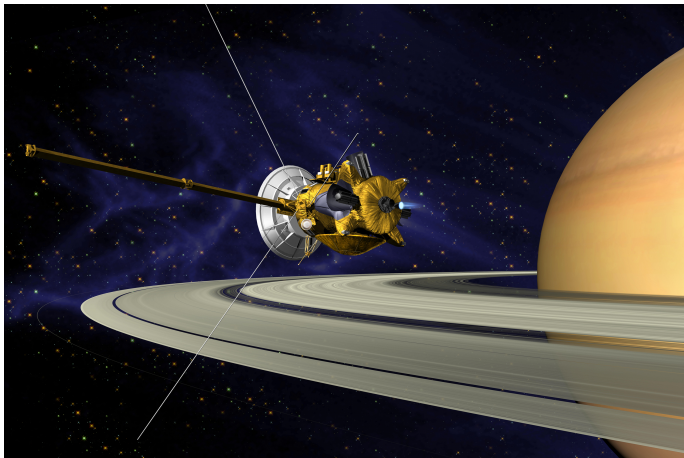


Figura: Inserção da sonda Cassini-Huygens na órbita de Saturno

- A sonda espacial Cassini–Huygens, lançada em 1997, possui dois gravadores de voo, cada um com 2,5 Gb de memória feita a partir de componentes de prateleira;
- Graças ao uso de EDCs, os números de erros em um (simples) ou em dois (duplo) bits por palavra eram conhecidos;
- Com o código utilizado, erros simples eram corrigíveis, erros duplos eram detectáveis, mas não corrigíveis;
- Por dia, eram reportados cerca de 280 de erros simples;
- Em 06/nov/1997, esse número aumentou em mais de quatro vezes. Foi atribuído a um evento de partículas solares detectado por um satélite.

- O código mais simples para detecção¹ de erros é utilizando o método da paridade – adiciona-se um bit de paridade ao fim bloco de bits transmitidos;
- O funcionamento é bem simples: o valor do bit de paridade será 0 ou 1 de forma que o número total de bits 1 transmitidos naquele bloco seja par (paridade ímpar) ou ímpar (paridade par);
- Exemplo de paridade ímpar: 000100110111 se torna 000100110111**1**;
- Exemplo de paridade par: 000100110111 se torna 000100110111**0**;
- Paridade somente consegue detectar erro em um único bit (erro simples)! Por isso, só é utilizado em aplicações onde a BER para erros simples é baixa e a BER para erros duplos é praticamente nula.

¹Mas não é aplicável para correção de erros!